DOCUMENT RESUME

ED 089 792 95 IR 000 517

AUTHOR	Peloquin, Paul V.
TITLE	1500 System Geometric Dictionary.
INSTITUTION	Pennsylvania State Univ., University Park. Computer-Assisted Instruction Lab.
SPONS AGENCY	Office of Education (DHEW), Washington, D.C.
REPORT NO	PSU-CAI-16
PUB DATE	30 Jun 68
CONTRACT	OEC-5-85-074
NOTE	22p.
EDRS PRICE	MF-$0.75 HC-$1.50 PLUS POSTAGE
DESCRIPTORS	Computer Assisted Instruction; *Computer Graphics; *Computer Programs; *Computer Science; Program Descriptions; *Programing
IDENTIFIERS	Cathode Ray Tube Terminals; Computer Assisted Instruction Laboratory; CRT Terminals; *Geometric Dictionary; IBM 1500 Computer System; Pennsylvania State University

ABSTRACT

A general description is provided of the "geometric dictionary," a graphic display aid, used by the Computer-Assisted Instruction Laboratory at the Pennsylvania State University. The purpose of the description is to enable the reader to duplicate and use the dictionary on any cathode ray tube terminal of the IBM 1500 system. The major advantages and disadvantages of the geometric dictionary are first discussed and a brief general overview given. Following this, details are presented regarding the steps of sketching, coding, and entering which are involved in the construction of line drawings. Information is also provided about keyboard functions, dictionary changes and the inclusion of text, blank characters, and shading. (PB)

ED 089792

C·A·I

COMPUTER ASSISTED INSTRUCTION LABORATORY

COLLEGE OF EDUCATION · CHAMBERS BUILDING

THE PENNSYLVANIA STATE UNIVERSITY · UNIVERSITY PARK, PA.

1500 SYSTEM GEOMETRIC DICTIONARY

PAUL V. PELOQUIN

June 30, 1968

Report No. R-16

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

IR 000 517

Note to accompany the Penn State Documents.

In order to have the entire collection of reports generated by the Computer Assisted Instruction Lab. at Penn State University included in the ERIC archives, the ERIC Clearinghouse on Educational Media and Technology was asked by Penn State to input the material. We are therefore including some documents which may be several years old. Also, so that our bibliographic information will conform with Penn State's, we have occasionally changed the title somewhat, or added information that may not be on the title page. Two of the documents in the CARE (Computer Assisted Remedial Education) collection were transferred to ERIC/EC to abstract. They are Report Number R-35 and Report Number R-50.

Jack McCall: ERIC/EM

1500 SYSTEM GEOMETRIC DICTIONARY

Paul V. Peloquin

Introduction

This is a general description of the geometric dictionary used by the Computer Assisted Instruction Laboratory at The Pennsylvania State University. It is hoped that this introduction will enable readers to duplicate and use this dictionary on any cathode ray tube (CRT) terminal of the IBM 1500 system. Only an elementary knowledge of certain Coursewriter II instructions will be assumed throughout this paper. (The reader is referred to the following sections of the Coursewriter II Manual [IBM, 1967]: Part I, pages 2-12; Part II, pages 7-15, 58-66.)

There are at least four advantages in using this geometric dictionary. First, the geometric dictionary uses less core storage than a graphic set, yet there is no limit to the number of line drawings which can be made. Because the components of a dictionary are small, they become more general and may be used in many different combinations. By analogy, a graphic set may be equated to a vocabulary of 64 words, while the geometric dictionary may be equated to an alphabet of 128 letters. The geometric dictionary may be equated and used within various courses thereby conserving core storage. Second, the necessity of keypunching each and every one of the line drawings dot by dot is eliminated. Third, the geometric dictionary allows the author to construct, on-line, the graphics for his course. This ability allows him to instantly see the line drawing as it is entered and make necessary changes or corrections. Fourth, words and line drawings can be combined without the one column gap necessary when using a graphic set.

There are, of course, some disadvantages as well. The geometric dictionary has been designed for producing moderate and large-sized line drawings. Except for some gross shading, such as "blackened" areas and hachures, the figures produced with this dictionary have been only line drawings. Some constraints are placed upon the drawing by the availability and the nature of the line segments. This restriction necessitates careful planning, but with the help of the IBM 1500 Instructional Display Planning Guide sheets it does not

constitute a serious problem. Usually the addition of an extra character into the dictionary or a bit of programing ingenuity will overcome problems in producing still or dynamic line drawings.

General Description

The geometric dictionary consists of line segments entered as dictionary characters which may be manipulated as such. Throughout this description the characters appearing on the terminal keyboard and on the CRT under the system dictionary, whether letters, symbols, or numbers, will be referred to as "associated keyboard characters." The characters containing the line segments of the geometric dictionary will be referred to as geometric characters, and the line parts contained within each of the geometric characters will be referred to as line segments, whether straight line segments, arcs, or special characters. Since we are dealing with dictionary characters, once the geometric dictionary has been called by a dictionary change the geometric dictionary has all the operating characteristics and functions of the system dictionary. A particular line segment is called and displayed on the screen by entering the associated keyboard character in a display text (DT) or display text insert (DTI) instruction. For example:

DT 12,10///*1aaaaa*b*b78 a a a a a 78*b*baaaaa*e

After a dictionary change (denoted by *1) is made, this instruction will display the "a" as a horizontal straight line and the numbers "78" as the left and right half of a small circle respectively. The backspace function (b*) has been used to superimpose lines.

Procedure

Sketching

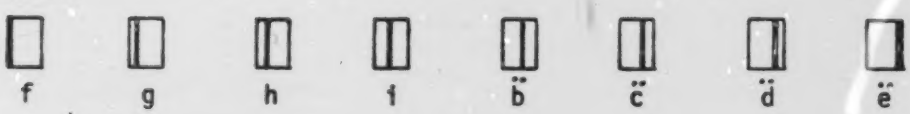
The procedure involved in the construction of a line drawing requires three steps: sketching, coding, and entering. First, a sketch of the line drawing is made on an Instructional Desplay Planning Guide. The sketches should be rather simple and should be constructed of those line segments that the programer knows are available in the dictionary. On the three pages following this description of the first step are reference pages indicating the

orientations of the line segments that are presently available to the programer. It is recommended that these reference pages be reproduced in a transparent form in order that the programer may overlay the available line segments on his sketch for comparison. The reader is reminded that each geometric character may be used independently. Thus if the third (and middle) character of a 30° line is needed, it may be used independently of the other four geometric characters which make up the completed line. If the programer cannot match his sketch with an available line, he should select the closest approximation and revise his sketch accordingly. A limited number of special characters may be inserted into the geometric dictionary if the programer finds that their omission seriously handicaps him. Additional flexibility can be gained by the use of the keyboard functions such as space, backspace, index, reverse index, and dictionary change. Superimposition, offsetting by a half-line, the display of text and geometric figures in juxtaposition or superimposition, the display of only half of a geometric character, and shading are some facilities gained by the use of standard keyboard functions. With a little practice, the programer's familiarity with the lines available in the dictionary should grow to a point where he will be able to produce sketches which require no lines that are not already in the dictionary. In anticipation of the second step, the programer may wish to make mental or written notes on the line segments he intends to use in constructing the line drawing.

REFERENCE SHEET FOR STRAIGHT LINE SEGMENTS

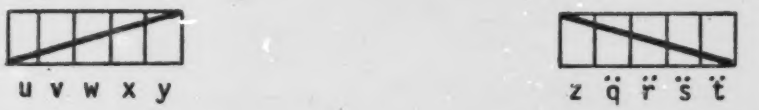
Horizontals

Verticals



Positive	Negative

15° Lines



30° Lines

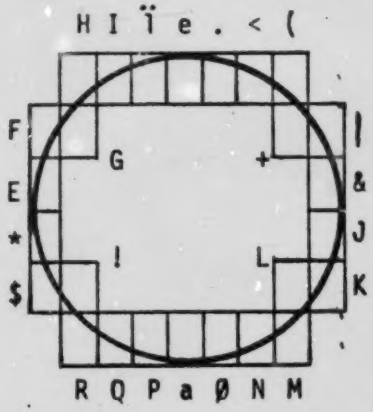
45° Lines

60° Lines

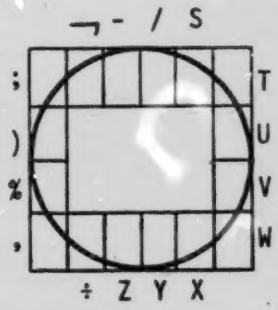
75° Lines

REFERENCE SHEET FOR CURVED LINE SEGMENTS

6 Characters Tall



4 Characters Tall



2 Characters

1 Character

1/2 Character

1 Row Tall

(2 in one Character)

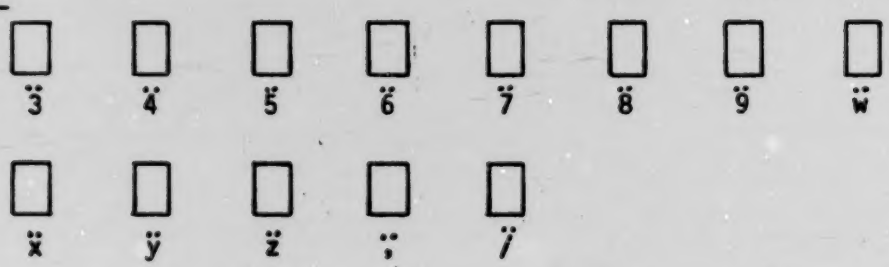
REFERENCE SHEET FOR SPECIAL AND BLANK CHARACTERS

Half Horizontals

Large Dot

Cursor

Blanks



Coding

The second step in constructing a figure with the geometric dictionary may take place once the sketch on the Instructional Display Planning Guide contains no lines that are not in the geometric dictionary. There are two methods by which this step may be accomplished. It is suggested, however, that the long form be used when learning to handle the dictionary and also to insure greater accuracy in joining lines.

Long Form. On the pages following the description of step two are the enlargements of the geometric characters shown on the preceding reference pages. (These enlargements may also be used for the original key punching of the dictionary.) These enlarged geometric characters permit close examination of the line segments to assure the programer that the lines he has chosen will assume the desired positions on the screen. Note that each small square on the enlargement represents one "dot" on the CRT. Associated with each geometric character is its number and the keyboard character which is used for encoding the geometric character, and thus the line segment. The final sketch should now be divided into sections, usually two rows in height (the height of one geometric character). The programer then writes a DT instruction covering those two or more rows. Then, after changing to the graphic dictionary, he encodes the associated keyboard character (or keypunch character) for each geometric character desired. It is suggested that the coding be written first on a coding form rather than entering directly on-line.

Short Form. If there is some familiarity with the geometric dictionary, and if there is no concern about lines in adjacent geometric characters matching, the short form is easier and faster to use. On the three reference pages preceding this section, the associated keyboard characters are written beside, above, or below the geometric characters to which they refer. One simply chooses the line segment and geometric character he wishes, then encodes the associated keyboard character.

Entering

The third step involves the on-line entering of the coded line drawing. This step may, of course, be altered by entering the coding on cards. The

advantage of entering on-line is that the programer may see the portions of the drawing as he enters it, thus allowing him to correct or change the coding and the drawing.

Special Considerations

Keyboard Functions

The keyboard functions such as space, carriage return, upshift, downshift, backspace, index, reverse index, and cursor display retain their original properties, but some now have special importance for the manipulation of line drawings.

Upshift, downshift and alternate code. The upshift, downshift, and alternate code functions will enable the programer to display three distinctly different geometric characters from one keyboard character. For example, down shifted (or lower case) "B," up shifted (or upper case "B," and alternate coded "B," will display a horizontal line, a small portion of a circle, and a vertical line respectively.

Backspace. As previously mentioned, the backspace function is of great importance to the geometric dictionary. Backspacing over a geometric character previously encoded within the same DT instruction will allow any number of superimpositions which can be effected short of exceeding the buffer length.

Index and reverse index. The index and reverse index functions allow line segments to be positioned one or more rows below or above the two rows currently being encoded. The greatest benefit achieved here is that a geometric character may be positioned a half character (one row) offset from its adjacent characters. The horizontal line segments in the lower half of the geometric character, for example, are constructed by using the index function, then displaying a line segment which in reality is in the upper half of the geometric character. Further, the programer can display only one half of a geometric character (and the portion of the line segment included) by allowing the geometric character to overlap one row above or below the limit of the display area stated in the parameters of the DT instruction. Only the half of the character within the display area will be shown. When he does not wish to truncate a geometric character, however, the programer is reminded that the use of the index or reverse index functions will require the number of rows used to be stated in the parameters of the DT instruction.

Dictionary Change and Inclusion of Text

Assuming that the geometric dictionary has been registered as usable dictionary number one (as opposed to the standard dictionary) for a course, the programer must include a dictionary change in every DT or DTI instruction using the geometric dictionary. Every new instruction assumes that the standard dictionary will be used unless otherwise specified. There is, however, no reason why a programer cannot change to the standard dictionary to enter some textual message in the middle of a line drawing, then return again to the geometric dictionary. In this manner text may be superimposed over geometric characters with no erasure of the line drawing.

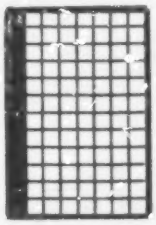
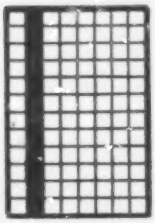
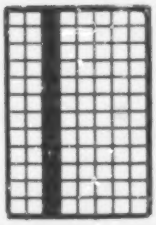
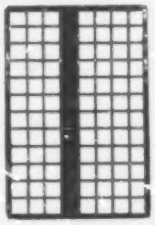
Blank Characters

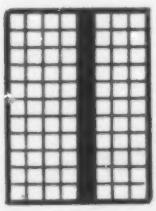
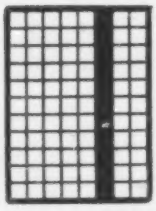
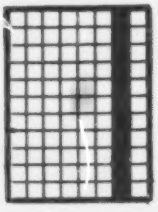
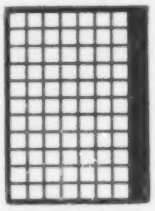
In the geometric dictionary the characters numbered 51 to 63 and 65 to 67 inclusive are unused. If, for some reason, the programer finds the addition of another character absolutely necessary, he may add the character by the usual means. Character number 64 should always remain blank since it is controlled by the space bar.

Shading

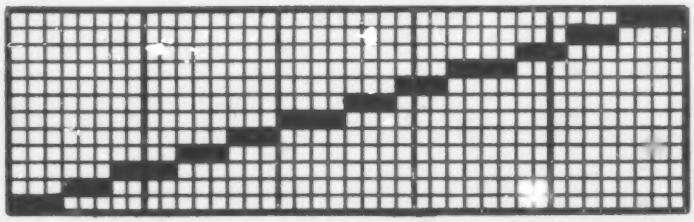
The only shading used has been "blackening" of a rectangle or hatching of some of the more regular figures. The "blackening" process is achieved by the superimposition of the vertical lines. Hachures are a bit more complex. Utilizing any of the straight line segments, the programer may superimpose them over the figure while taking care that the hachures do not protrude beyond the figure. The enlarged forms are helpful guides for this purpose.

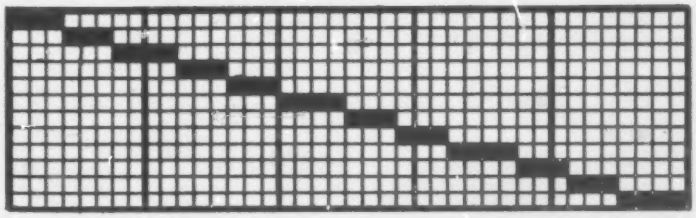
VERTICALS

Character number.....	006	007	008	009
				
029 Keypunch.....	f	g	h	i
On-Line Input.....	f	g	h	i

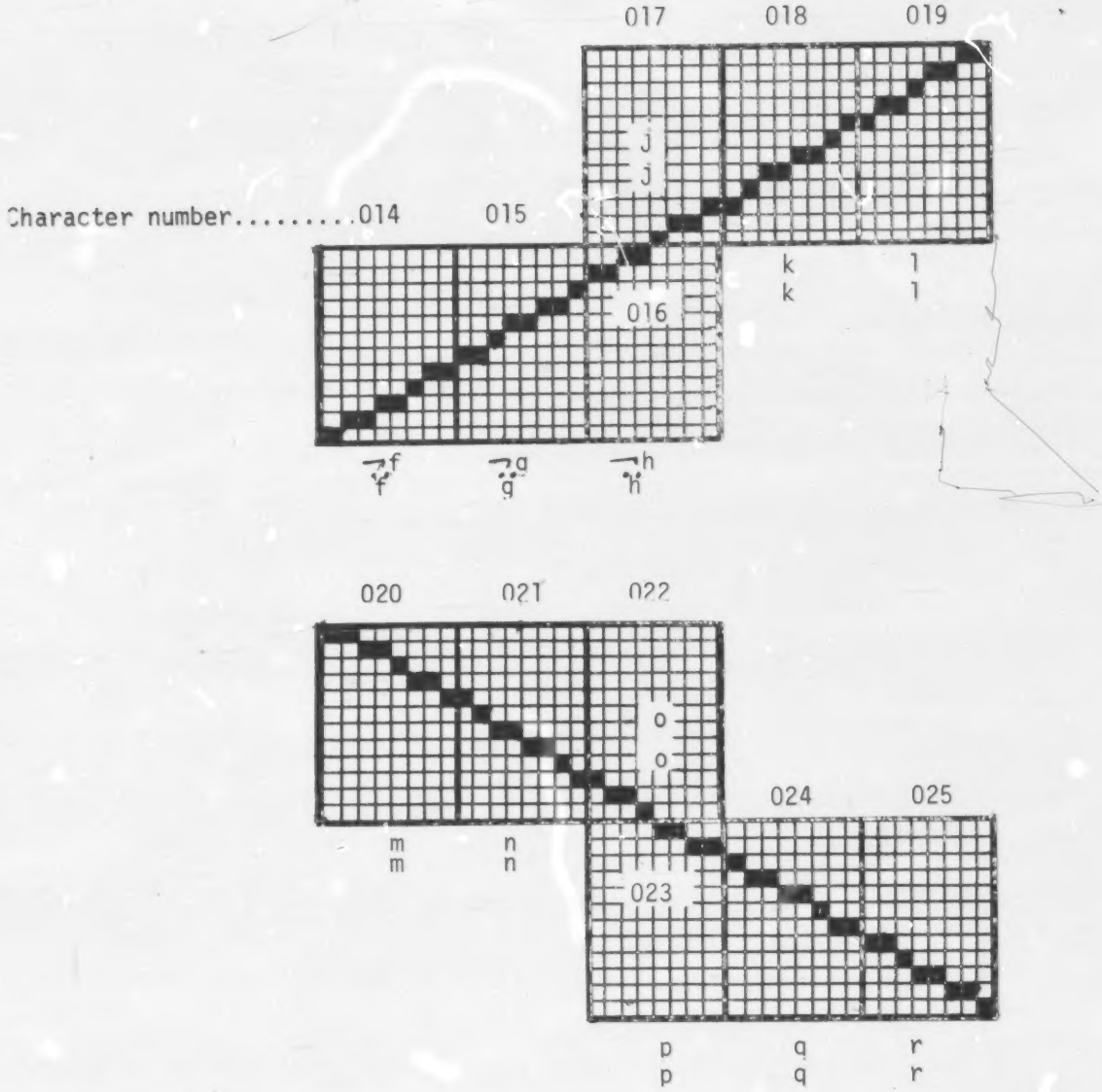
010	011	012	013
			
b	c	d	e

15° LINES

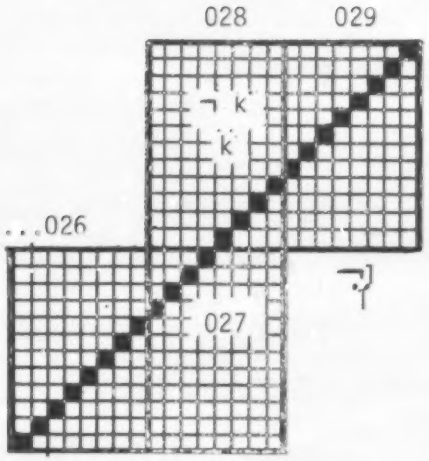
Character number.....	036	037	038	039	040
					
029 Keypunch.....	u	v	w	x	y
On-Line Input.....	u	v	w	x	y

	041	042	043	044	045
					
	z	¬ q	¬ r	¬ s	¬ t
	z	q	r	s	t

30° LINES

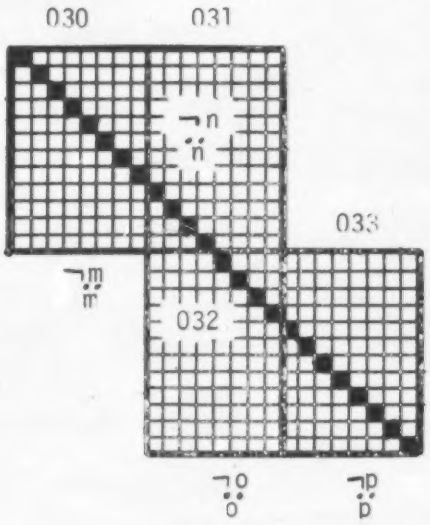

017
018
019
J
j
Character number.....014
015
k
k
l
l
016
f
f
g
g
h
h
020
021
022
o
o
024
025
m
m
n
n
023
p
p
q
q
r
r

45° LINES



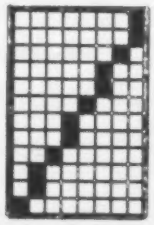
Character number.....026

029 Keypunch..... i j
On-Line Input..... i j



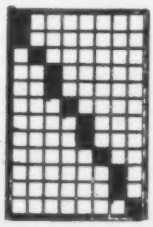
60° LINES

Character number.....034



029 Keypunch.....s
On-Line Input.....s

035

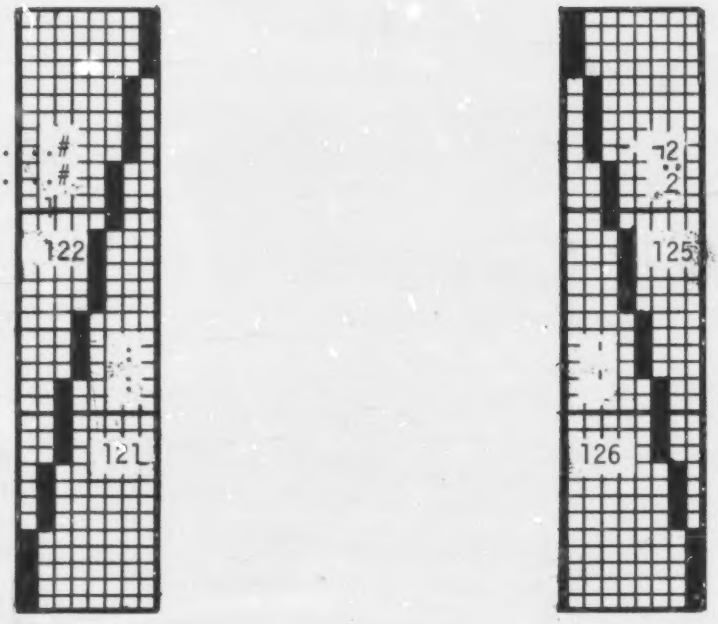


t
t

75° LINES

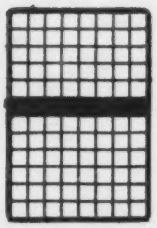
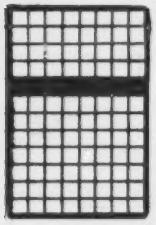
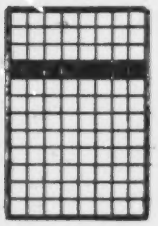
Character number.....123 124

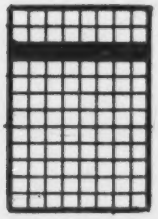
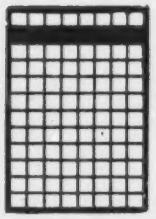
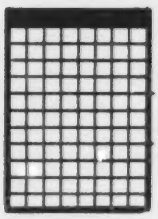
029 Keypunch.....
On-Line Input.....

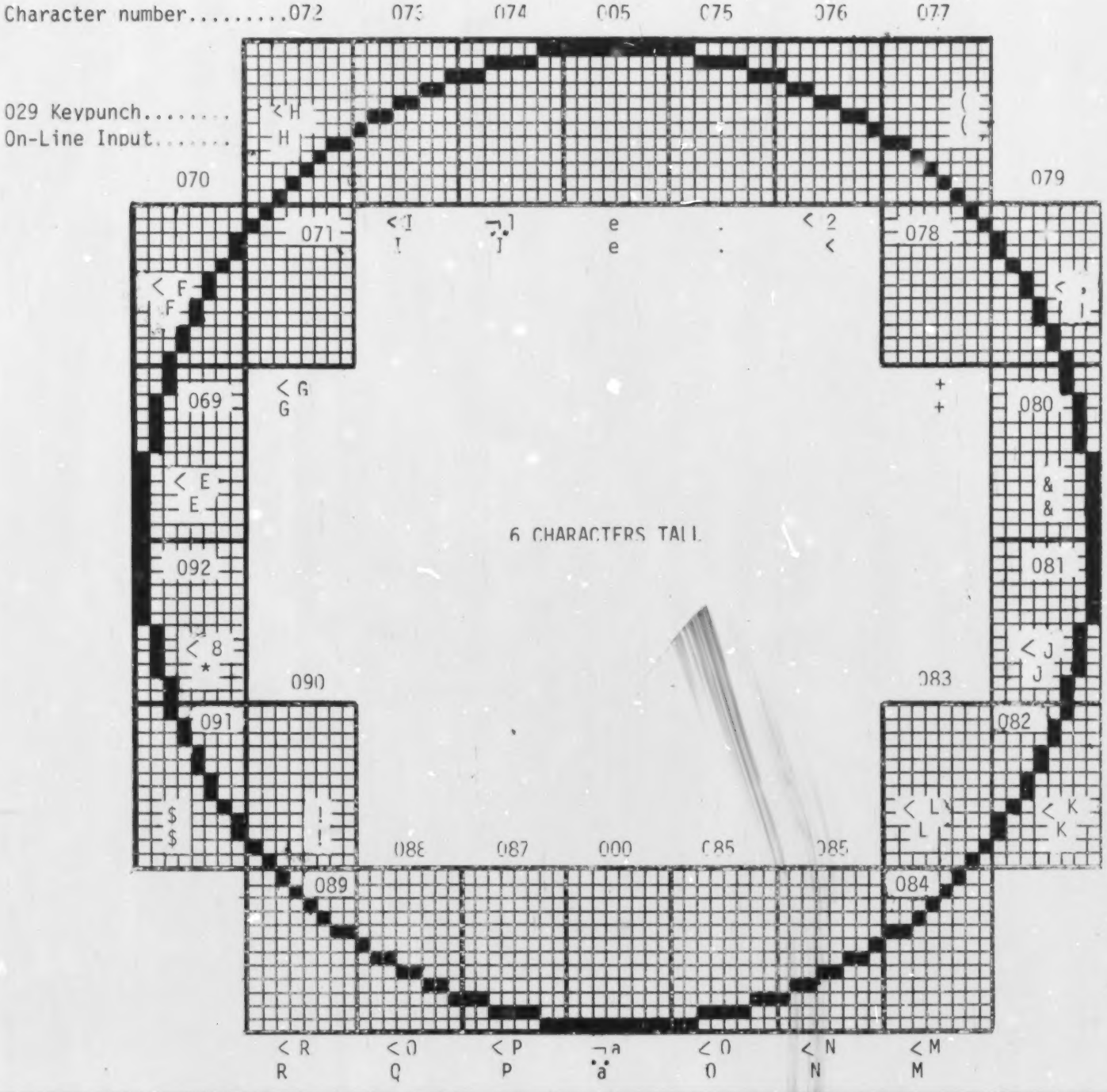


9 =
9 =

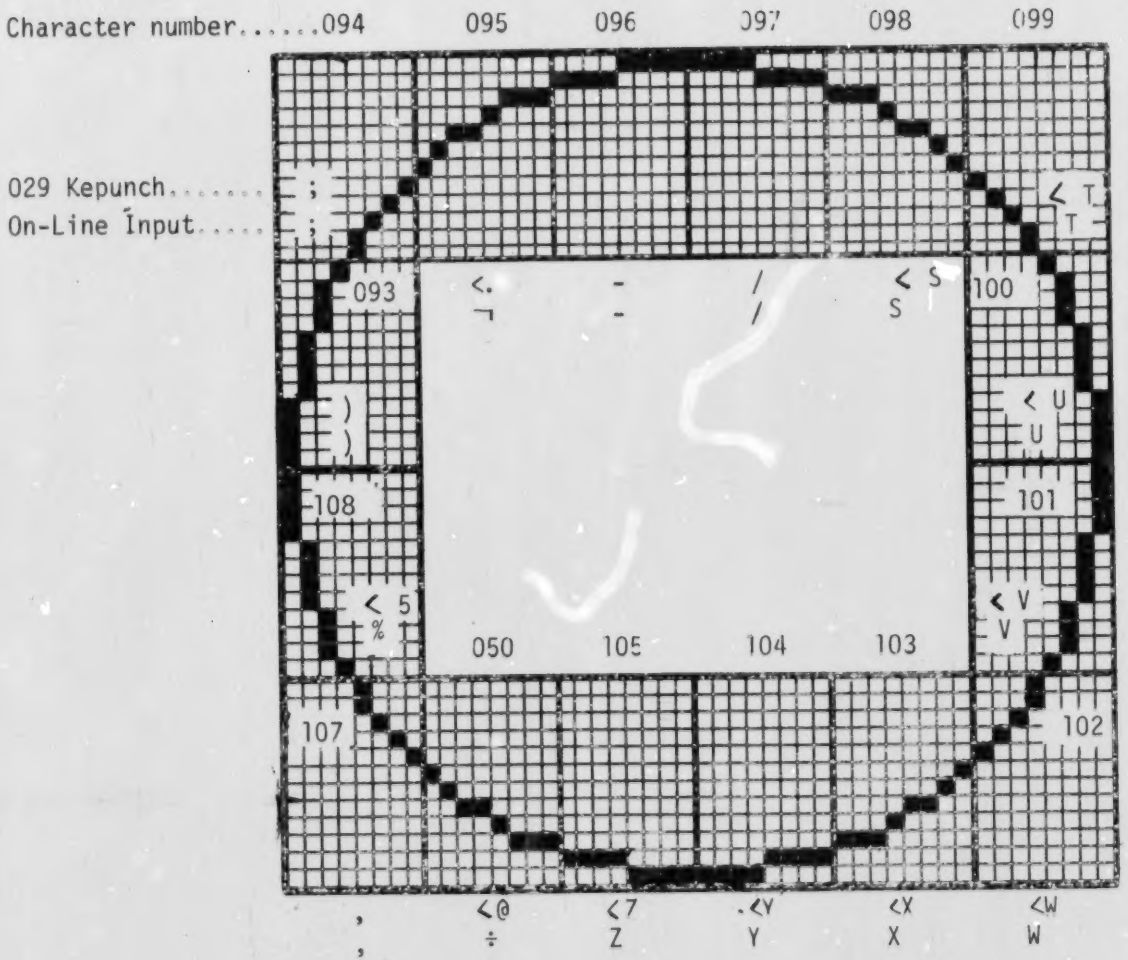
HORIZONTALS

Character number	000	001	002
			
029 Keypunch.....	¬a	a	b
On-line input.....	ä	a	b

003	004	005
		
c	d	e
c	d	e


Character number.....072 073 074 075 076 077
029 Keypunch.....
On-Line Input.....
070
071
079
078
069
080
092
081
090
083
091
082
089
084
6 CHARACTERS TALL

4 CHARACTERS TALL



Character number.....109 110 111

029 Keypunch..... - < 1 ?
On-Line Input..... - > ?

114 113 112

2 1 0
2 1 0

(zero)

1 CHARACTER TALL

115 116

3 4
3 4

118 117

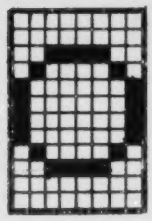
6 5
6 5

119 120

7 8
7 8

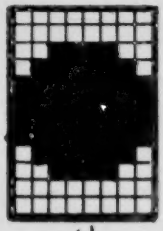
1 CHARACTER WIDE

068



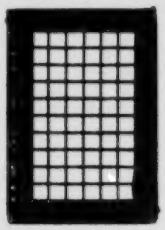
<D
D

Character number..... 127



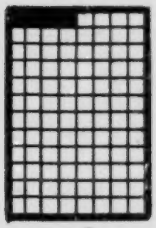
029 Keypunch..... <'
On-Line input..... "

106



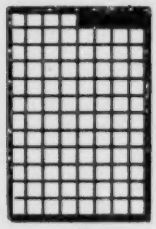
¬;
:

048



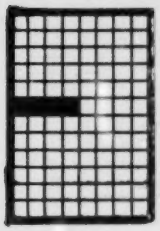
¬C
:
0 (zero)

049



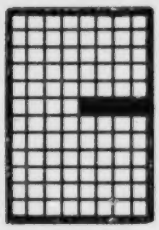
@
X (times)

046



¬u
:
u

047



¬v
:
v

Reference

IBM Systems Development Division, Product Publications. 1500 Coursewriter II Author's Guide. San Jose, California: IBM, 1967, 2 parts.